UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/052,999	11/02/2001	Partha P. Tirumalai	SUN-P7133-RA	1267

57960 7590 10/03/2006

SUN MICROSYSTEMS INC.
C/O PARK, VAUGHAN & FLEMING LLP
2820 FIFTH STREET
DAVIS, CA 95618-7759

EXAMINER	
FOWLKES, ANDRE R	
ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 10/03/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/052,999	**Applicant(s)** TIRUMALAI ET AL.	
	Examiner Andre R. Fowlkes	**Art Unit** 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 June 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-17* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed 6/16/06.

2. Claims 1-17 are pending. Claims 1, 8 and 13 have been amended.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Santhanam, U.S. Patent No. 5,704,053 in view of Wu, et al., (Wu), U.S. Patent Publication No. 2003/0066061.

As per claim 1, Santhanam discloses **a method for generating code to perform anticipatory prefetching for data references,** (col. 3:47-49, "The current invention provides a new compiler for such a processor that facilitates efficient insertion of explicit data prefetch instructions into loops within application programs"), **comprising:**

- **receiving code to be executed on a computer system; analyzing the code to identify data references to be prefetched** (col. 3:50-51, "The compiler uses ... analysis (techniques) to determine data prefetching requirements"),

- **calculating a prefetch ahead distance... wherein the prefetch ahead distance indicates the number of loop iterations ahead to prefetch for**, (col. 3:50-63, "The current invention provides a new compiler for such a processor that facilitates efficient insertion of explicit data prefetch instructions into loops within application programs. The compiler uses simple subscript expression analysis (and the subscripts of a loop determine how many loop iterations are performed) to determine data prefetching requirements (i.e. the prefetch ahead distance, indicating the number of loop iterations ahead to prefetch for) ... Cache line reuse patterns across loop iterations are recognized to ... (optimize the number of) prefetch instructions"),

- **inserting prefetch instructions into a preceding basic block of the code in advance of the identified data references based upon** code analysis (col. 3:51-53, "Analysis and explicit data cache prefetch instruction insertion are performed by the compiler", and prefetch instructions are always inserted into the code preceding the identified data reference to allow sufficient time for data to be prefetched),

- **wherein inserting prefetch instructions involves inserting multiple prefetch instructions for a given cache line** (col. 6:61-62, "the system is issuing a redundant (prefetch) instruction(s) to the memory system to retrieve the same cache line"),

- **wherein inserting the prefetch instructions involves:**

- attempting to calculate a stride value for a given data reference within a loop (col. 6:3-5, "The compiler can predict (by attempting to calculate a stride value) which data (reference) is needed in advance for loops that access array elements in a regular fashion"),

- if the stride value cannot be calculated, setting the stride value to a default stride value (col. 14:48-49, "(if the stride cant be calculated), then substitute some fixed constant, C"),

- inserting a prefetch instruction to prefetch the given data reference for a subsequent loop iteration based on the stride value (col. 6:5-8, "The compiler can then insert prefetch instructions into loops such that array elements that are likely to be needed in future loop iterations are retrieved from memory ahead of time"),

- wherein the stride value is constant for some (but not necessarily all) loop iterations (col. 2:25-28 ,"because the analysis is done at the source code level, it is difficult to estimate the prefetch iteration distance (PFID) (in this situation the stride value is constant for some but not necessarily all loop iterations), i.e. the PFID used is always one loop iteration (i.e. default prefetch distance)").

Although Santhanam discloses a prefetch ahead distance, Santhanam doesn't explicitly disclose that the prefetch ahead distance includes:

- the ratio of outstanding prefetches to the number of prefetch streams.

However, Wu in an analogous environment, discloses that the prefetch ahead distance includes:

- **the ratio of outstanding prefetches to the number of prefetch streams**

(Para. 82:5-10, "If the frequency ratio of load instruction (i.e. the ratio of outstanding prefetches/loads to the number of prefetch/load streams) exceeds a predefined threshold, ...(the) load instruction (i.e. prefetch instruction) can be (optimized)"). Further, Wu discloses that "various factors are considered when identifying potential candidates for ... (optimization), e.g. whether a potential candidate has a predictable data flow (i.e. loop characteristics), whether it is on a critical path, etc." This is analogous to applicant's arguments at p. 8:20-25, "the present invention examines ... the outstanding prefetches value and loop characteristics."

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Wu into the system of Santhanam to have a optimization of code based on a prefetch ratio. The modification would have been obvious because one of ordinary skill in the art would have wanted to use well known and well document code analysis & optimization techniques to increase the effectiveness of computer systems which utilize prefetch optimizations. Furthermore, it is well known and well documented in the art, that value profile analysis, often including ratios, is a useful technique to determine when it is appropriate to perform several types of optimizations, including prefetching. For example, Waldspurger et al., U.S. Patent No. 6,961,930 (art made of record), describes at col.

15:50-54 that, "Value profiles can also be exploited to drive ... optimizations, including code specialization, software speculation and prefetching."

As per claim 2, the rejection of claim 1 is incorporated and further, Santhanam discloses **allowing a system user to specify the default stride value** (col. 13:39, "Estimating the average loop iteration latency").

As per claim 3, the rejection of claim 1 is incorporated and further, Santhanam discloses that **calculating the stride value involves:**

- identifying an induction variable for the stride value (col. 11:23, "Identify simple basic loop induction variables"),

- identifying a stride function for the stride value and calculating the stride value based upon the stride function and the induction variable (col. 17:54-60, "a net loop increment of eight, and the element size of "A" is 8-bytes, this is a large stride equivalence class, assuming a 32-byte cache line size (8.times.8 bytes=64 bytes)>32 bytes").

As per claim 4, the rejection of claim 1 is incorporated and further, Santhanam discloses that **inserting the prefetch instruction based on the stride value involves:**

- calculating a prefetch cover distance by dividing a cache line size by the stride value (col. 15:64-67, "When the memory stride is <=cache line size, B(i) is considered to be in the same cluster as B(i+1), and therefore omitted for prefetch consideration (i.e. the prefetch cover distance is calculated based on the cache line size and stride value)", and col. 17:54-66, "(Because the loop has) a net loop increment of eight, and the element size of "A" is 8-bytes, this is a large stride equivalence class, assuming a 32-byte cache line size (8.times.8 bytes=64 bytes)>32 bytes. All eight references to " A" are placed into the same cluster because they exhibit group spatial locality, and no group temporal locality. The cluster leader is the reference to A[i+7], and the span of the cluster is 64-bytes (i.e. &A[i+7]-&A[i]). If the prefetch memory distance was computed earlier to be 128-bytes, i.e. corresponding to a prefetch iteration distance of two, it is only necessary to insert three prefetch instructions to account for the entire span of this 8-member cluster.") ,

- **calculating a prefetch ahead distance as a function of a prefetch latency, the prefetch cover distance and an execution time of a loop** (col. 7:11-18, "The memory address is determined based on the number of loop iterations in advance (i.e. the prefetch iteration distance or PFID) that data items need to be prefetched to fully hide the time required to service potential data cache misses. The PFID is determined taking into account the nature of the loop body instructions (i.e. execution time of the loop and the prefetch cover distance) and characteristics of the target processor and memory system (i.e. the prefetch latency and prefetch cover distance)"),

- **calculating a prefetch address by multiplying the stride value by the prefetch cover distance and the prefetch ahead distance and adding the result to an address accessed by the given data reference** (col. 7:11-18, "The memory address is determined based on the number of loop iterations in advance (i.e. the prefetch iteration distance or PFID) that data items need to be prefetched to fully hide the time required to service potential data cache misses. The PFID is determined taking into account the nature of the loop body instructions and characteristics of the target processor and memory system.").

As per claim 5, the rejection of claim 1 is incorporated and further, Santhanam discloses that analyzing the code involves:

- **identifying loop bodies within the code; identifying data references to be prefetched from within the loop bodies** (col. 8:30-35, "One important feature of the invention identifies loops and access patterns to allow a determination of how many cycles are devoted to loop iterations, and therefore allows insertion of the prefetch instruction to a location of an array that is sufficiently far in advance to make sure that the miss time is minimized.").

As per claim 6, the rejection of claim 5 is incorporated and further, Santhanam discloses that analyzing the code to identify data references to be prefetched involves **examining a pattern of data references over multiple loop iterations** (col. 14:6-10, "Now, it is also necessary to address the issue of loops that have internal branches. The minimum loop iteration latency for such loops is estimated by using previously collected execution profile information, which indicates the execution count for each basic block in the loop body.").

As per claim 7, the rejection of claim 1 is incorporated and further, Santhanam discloses that analyzing the code involves **analyzing the code within a compiler** (col. 3:47-49, "The current invention provides a new compiler for such a processor that facilitates efficient insertion of explicit data prefetch instructions into loops within application programs").

As per claims 8-12, this is a computer readable medium/product version of the claimed method discussed above, in claims 1-7 , wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see Santhanam, col. 3:47-49 and Wu, Para 25:1-26:11 and 82:1-10.

As per claims 13-17, this is an apparatus version of the claimed method discussed above, in claims 1-7 , wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see Santhanam Fig. 1 item 10, "computer architecture" and associated text and Wu, and Wu, Para 25:1-26:11 and 82:1-10.

Response to Arguments

5. Applicants arguments have been considered but they are not persuasive.

In the remarks, the applicant has argued substantially that:

1) The Wu art teaches value specialization, in contrast to the prefetching optimization done by the instant application, at p. 8:7-21.

Examiner's response:

1) As an initial matter, Santhanam is used to disclose performing prefetch operations. The Wu art is used to disclose using a value profile, such as a ratio to determine when it is appropriate to perform a compiler optimization (e.g. pre-loading a value in advance of a load instruction). More specifically, Wu uses a value profile to determine when it is appropriate to perform the an optimization, at Para. 25:1-27:5, "a program is examined to identify potential candidates for value specialization (i.e. optimization)... Based on their value profiles, potential candidates are evaluated ... (and potentially) added to a group of selected candidates". It is well known and well documented in the art, that value profiles are used to determine when it is appropriate to perform several other types of optimizations, including prefetching. For example, Waldspurger et al., U.S. Patent No. 6,961,930 (art made of record), describes at col. 15:50-54 that, "Value profiles can also be exploited to drive ... optimizations, including code specialization, software speculation and prefetching." Accordingly, value profiles are used for both predicting/value specialization and prefetching.

In the remarks, the applicant has argued substantially that:

2) Santhanam does not disclose using a prefetch ahead distance to determine how many loop iterations ahead to prefetch for, at p. 9:3-4.

Examiner's response:

2) The examiner disagrees with applicant's characterization of the applied art. Santhanam discloses calculating a prefetch ahead distance, wherein the prefetch ahead distance is a metric which indicates the number of loop iterations ahead to prefetch for, at col. 3:50-63, "The current invention provides a new compiler for such a processor that facilitates efficient insertion of explicit data prefetch instructions into loops within application programs. The compiler uses simple subscript expression analysis (and the subscripts of a loop determine how many loop iterations are performed) to determine data prefetching requirements (i.e. the number of loop iterations ahead to prefetch for) ... Cache line reuse patterns across loop iterations are recognized to ... (optimize the number of) prefetch instructions," as described in the above art rejection.

In the remarks, the applicant has argued substantially that:

3) Santhanam does not teach calculating a prefetch ahead distance, wherein the prefetch ahead distance includes the ratio of outstanding prefetches to the number of prefetch streams, at p. 9:9-12.

Examiner's response:

3) The examiner would like to clarify that Santhanam does in fact discloses calculating a prefetch distance, at col. 3:50-63, as described in the above art rejection. However, the Wu art is included because Santhanam does not explicitly disclose calculating the ratio of outstanding prefetches to the number of prefetch streams. Wu remedies this deficiency, rendering the instant invention as obvious in view of the Santhanam/Wu combination.

Conclusion

6. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andre R. Fowlkes whose telephone number is (571)

272-3697. The examiner can normally be reached on Monday - Friday, 8:00am-4:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571)272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

ARF

TUAN DAM
SUPERVISORY PATENT EXAMINER